CLAIMS:

1. A method for determining patterns in an input string of tokens, comprising steps of:

identifying extensible patterns in the input string;

creating an inexact tree for the input string, using the patterns identified; and

displaying a set of extensible patterns identified by the inexact tree;

wherein creating the inexact tree comprises creating nodes and edges, connecting the nodes,

wherein each node represents a subset of a string and each edge connects a lower order node to a higher order node;

wherein each subset comprises a pattern comprising extensible string; and

wherein each extensible string comprises at least one dot token.

2. The method of claim 1, further comprising receiving a parameter *k* specifying the minimum times an extensible pattern must occur in a sequence.

3. The method of claim 1, wherein the step of identifying patterns in the input string *B* comprises creating a rigid string *m'* from an extensible string *m*.

4. The method of claim 1 wherein the step of identifying patterns in the input string *B* comprises extracting a subset of tokens *b* from the input string *B*.

5. The method of claim 4 analyzing the subset of tokens *b* to determine whether the subset is compatible with the rigid string *m'*.

6. The method of claim 5 wherein if the subset b is compatible with the rigid string m' the subset and the rigid string are concatenated into a new rigid string m_t.

7. The method of claim 6 further comprising the step of running a routine for determining whether the concatenated string is non maximal with respect to its nodes of the same order.

8. The method of claim 7 further comprising removing each node form the tree that is non maximal with respect to its nodes of the same order.

9. The method of claim 8 wherein if the magnitude of the location list of the rigid string m' is equal to the magnitude of the location list of the subset of tokens b then the size of the collection of tokens B is reduced by removing the subset of tokens b determined in the step of extracting a subset of tokens from the input string.

10. The method of claim 9 wherein if the number of times the rigid string pattern repeats is greater than the minimum number of times an extensible pattern must occur in a sequence k, then the concatenated extensible string m_t is converted into a rigid string m'.

11. The method of claim 10 wherein the method of claim 1 is performed on the converted rigid string m'.

12. The method of claim 11 further comprising identifying a zone for each subsequence of tokens Z_r such that each occurrence of each pattern is fully contained within the zone of the rigid string $Z_{m'}$.

13. The method of claim 11 further comprising determining whether the rigid string m' is not maximal with respect to a string of tokens r that are returned from the determination of the routine.

14. The method of claim 13 wherein the result of the routine m' is added to a collection of maximal extensible patterns *Result*.

15. A system comprising:

an input/output device for receiving information including an input string; and

a processor for identifying extensible patterns; and

a memory for storing identified patterns and for storing the inexact suffix tree.

16. The system of claim 15 wherein the input/output device further comprising a CD ROM drive.

17. The system of claim 15 wherein the input/output device further comprises a network interface.

18. The system of claim 15 wherein the memory further comprises an operating system.

19. The system of claim 15 wherein the memory further comprises an application.

20. A program product for determining patterns in an input string of tokens, comprising instructions for:

identifying extensible patterns in the input string;

creating an inexact tree for the input string, using the patterns identified; and

displaying a set of extensible patterns identified by the inexact tree;

wherein creating the inexact tree comprises creating nodes and edges, connecting the nodes,

wherein each node represents a subset of a string and each edge connects a lower order node to a higher order node;

wherein each subset comprises a pattern comprising extensible string; and

wherein each extensible string comprises at least one dot token.